CrimsonCream Inc. Marketing Campaign Analysis Using Multi-Regression

Santiago Paiva
sap789@g.harvard.edu

STAT104 - Quantitative Methods for Economists
Harvard University

May 3, 2016

Abstract

We evaluated, using regression models, the performance of the Marketing campaign run in 2016 by CrimsonCream Inc. in three different cities: New York, Chicago, and Los Angeles. We implemented different diagnostic tools to test the accuracy of the data and to identify potential outliers in our data. We found the Backward Regression using AIC model to be the best performing model and that the Marketing campaign indeed yielded more sales.

Contents

1 Introduction

The market share has beginning to noticed a small decline and CrimsonCream Inc. decided to to embark on a promotion campaign in all three operating cities: New York, Chicago, Los Angeles to improve sales. A dataset of 1000 observations over a year was collected and we are interested in comparing the results of running a Marketing campaign while the overall national economic sentiment declined. Importantly, we want to evaluate the results of the campaign to assess whether the campaign helped increase sales or not. We hypothesize that the Marketing campaign (Promo) helped increase ice cream sales.

1.1 General Overview

In Table 1 we provide a general overview of the dataset we analyzed in our model which includes 15 variables. Our variable in question is `salespercap`, which we hope to estimate. The `promo` variable is a categorical variable with value = 0 from January to July when the Marketing campaign was not run, and with value = 1 from August to December when the Marketing campaign was run.

Variable	Obs	Mean	Std. Dev.	Min	Max
salespercap	1,000	6.21293	.8561946	3.51	9.02
ny	1,000	.339	.4736067	0	1
la	1,000	.322	.4674768	0	1
chi	1,000	.339	.4736067	0	1
price	1,000	.9993499	.1156039	.80021	1.19744
comp	1,000	1.254442	.1423781	1.00053	1.49911
economy	1,000	81.548	13.53554	65	100
day_year	1,000	182.066	104.7911	1	365
day_week	1,000	4.003	1.999247	1	7
weekend	1,000	.286	.4521155	0	1
newchain	1,000	.103	.304111	0	1
temp_ny	1,000	18.53488	27.50133	0	89.868
temp_chi	1,000	16.32716	25.14477	0	89.1891
temp_la	1,000	24.0966	35.33863	0	89.8935
promo	1,000	.496	.5002342	0	1

Table 1: Summary of the dataset

SALESPERCAP

	Percentiles	Smallest		
1%	4.165	3.51		
5%	4.745	3.64		
10%	5.075	3.86	Obs	1,000
25%	5.69	3.93	Sum of Wgt.	1,000
50%	6.23		Mean	6.21293
		Largest	Std. Dev.	.8561946
75%	6.71	8.94		
90%	7.225	8.97	Variance	.7330692
95%	7.615	8.99	Skewness	.062187
99%	8.47	9.02	Kurtosis	3.596262

Table 2: Interquartile overview of sales per capita

Figure 1 both show the density distribution of the total sales per capita of cups of ice cream and the breakdown of the distribution by `promo`. Distributions seems to be similar and they follow a Normal density distribution. Figure 2 shows a side by side box plot comparison of sales by promo and by non-promo. Figure 3 represents shows the distribution of the sales per capita as a function of ice cream price.

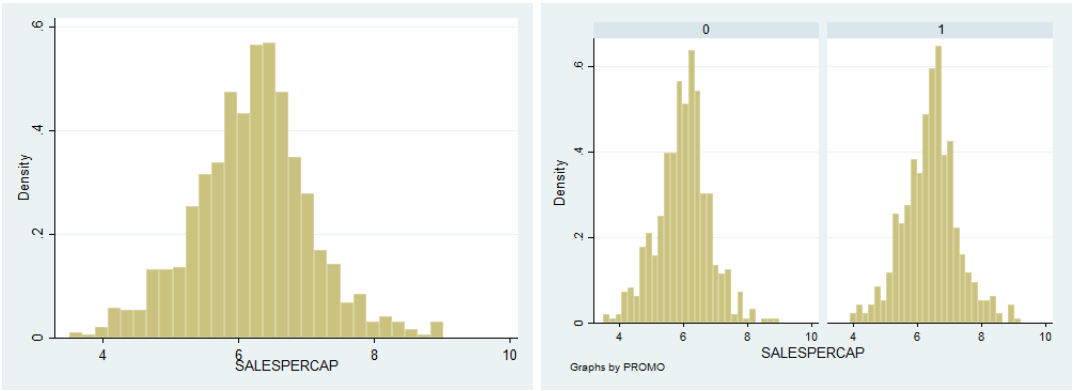

Figure 1: Density distribution of sales per capita. Left: Overall sales per capita. Right: Sales per capita by Promo (Marketing campaign)

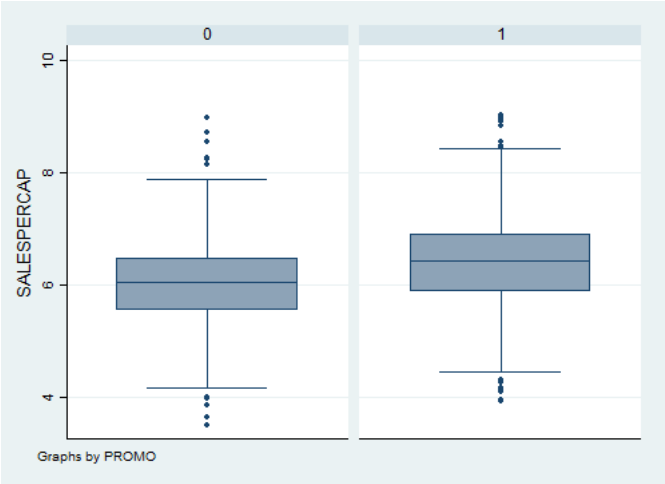

Figure 2: Promo Sales seem to perform better than non-promo

Figure 3: Sales per capita as a function of Price

1.2 Overview by City

Figure 4 shows the breakdown of the sales per capita distribution by operating city: New York, Los Angeles, and Chicago

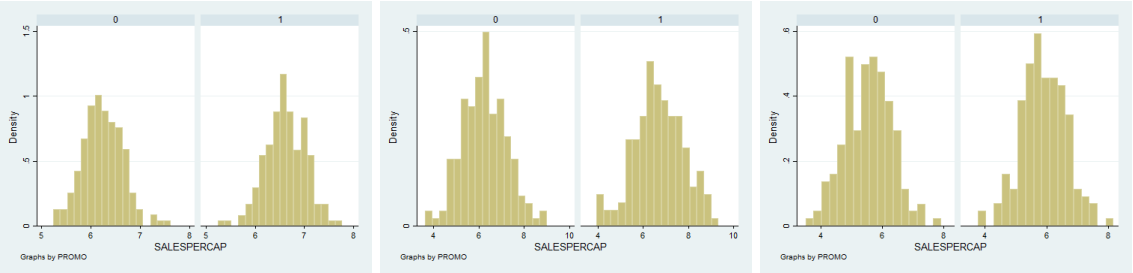

Figure 4: Sales per capita distribution by city. Left: New York. Middle: LA. Right: Chicago

2 Methods

2.1 Dataset & Software

The dataset analyzed in this paper was taken from the following source `http://people.fas.harvard.edu/~mparzen/stat104/icecream2016V1` which contains information about ice cream sales in 2016. The software and version used for the these analyses was Stata/MP 14.0

2.2 Diagnostic Tests

In order to account for all the technical problems that could arise in regression, we evaluated Multicollinearity (`vif`), Heteroscedasticity noise (`hettest`), Normality (`sktest`), Outliers (Residuals, Cook's Distance), and Non-linearity (`ovtest`) in our observations

2.3 Hypothesis Testing

The following Null hypothesis were taken into account for diagnostics:

- H_0 for Heteroscedasticity: noise is homoscedastic
- H_0 for Normality: data not normal
- H_0 for Non-linearity: transformation needed

The criteria for statistical significance for Hypothesis testing is $P < 0.05$

2.4 Regression Models

A total of four regression models were implemented in this paper to predict sales per capita:

1. Naïve Regression
2. Normal Backward Regression
3. Backward Regression using Adjusted R-Square
4. Backward Regression using AIC

3 Results

3.1 Naïve Regression Model

In our first regression model (Model I), we include all variables, we did not perform any diagnostics, transformations, or any modifications to the model, and we use `price` as baseline. Table 3 shows our first regression model

Source	SS	df	MS	Number of obs	=	1,000
				F(13, 986)	=	26.63
Model	190.286813	13	14.6374472	Prob > F	=	0.0000
Residual	542.049299	986	.549745739	R-squared	=	0.2598
				Adj R-squared	=	0.2501
Total	732.336112	999	.733069181	Root MSE	=	.74145

salespercap	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
la	-.1487033	.3869216	-0.38	0.701	-.9079878	.6105812
chi	-.475583	.1862478	-2.55	0.011	-.8410706	-.1100953
price	-.5562765	.2036489	-2.73	0.006	-.9559116	-.1566413
comp	.7316247	.165792	4.41	0.000	.4062791	1.05697
day_year	.0007815	.0008874	0.88	0.379	-.0009598	.0025228
day_week	.01557	.0193605	0.80	0.421	-.0224225	.0535625
weekend	.2539252	.0856144	2.97	0.003	.0859178	.4219327
newchain	-.0519348	.1004818	-0.52	0.605	-.2491177	.145248
temp_ny	.001728	.0027622	0.63	0.532	-.0036924	.0071484
temp_chi	-.001982	.0024568	-0.81	0.420	-.0068032	.0028393
temp_la	.0041203	.004697	0.88	0.381	-.0050969	.0133375
promo	.5549447	.1306492	4.25	0.000	.2985623	.8113272
economy	.0112662	.0088181	1.28	0.202	-.0060381	.0285705
_cons	4.495346	.96913	4.64	0.000	2.593552	6.397141

Table 3: Model I - The Naïve Regression Model

This first model is particularly bad. Both $R^2 = 0.25$ and Adjusted $R^2 = 0.25$ are significantly low, S_e is very high, and the Confidence Intervals describe which variables we do not need in the model: `ny`, `la`, `day_year`, `day_week`, `newchain`, `temp_ny`, `temp_chi`, `temp_la`, `economy`

3.2 Test for Correlation

We run a correlation test to identify which variables are highly correlated between each other that might influence our model. Table 4 shows the correlations of all the Xs in our regression model.

	ny	la	chi	price	comp	economy	day_year
ny	1.0000						
la	-0.4935	1.0000					
chi	-0.5129	-0.4935	1.0000				
price	-0.0101	0.0023	0.0078	1.0000			
comp	0.0334	-0.0233	-0.0105	-0.0324	1.0000		
economy	-0.0079	0.0112	-0.0031	-0.0444	-0.0014	1.0000	
day_year	0.0066	-0.0089	0.0022	0.0557	-0.0044	-0.9623	1.0000
day_week	0.0084	-0.0010	-0.0074	0.0024	0.0181	0.0013	0.0065
weekend	0.0096	-0.0099	0.0002	0.0126	-0.0252	-0.0004	0.0056
newchain	-0.2427	-0.2335	0.4732	0.0018	0.0192	-0.3795	0.4082
temp_ny	0.9416	-0.4647	-0.4829	-0.0152	0.0453	-0.0981	0.0761
temp_chi	-0.4652	-0.4477	0.9072	0.0123	0.0058	-0.1336	0.1056
temp_la	-0.4886	0.9899	-0.4886	0.0027	-0.0264	0.0091	-0.0067
promo	0.0078	-0.0116	0.0036	0.0355	0.0081	-0.9260	0.8668

Table 4: Correlation across all X variables

Coefficients of > 0.5 indicate potential Multicollinearity problem.

3.3 Test for Multicollinearity

After looking at the correlation of all the Xs in the model, we look for Multicollinearity, i.e., X variables highly related to each other with the Variance Inflation Factors (VIF) test in Stata. Table 5 shows variables that are highly related to each other

Variable	VIF	1/VIF
la	59.45	0.016821
temp_la	50.07	0.019974
economy	25.88	0.038646
day_year	15.68	0.063781
chi	14.14	0.070734
temp_ny	10.48	0.095412
promo	7.76	0.128836
temp_chi	6.94	0.144195
day_week	2.72	0.367357
weekend	2.72	0.367371
newchain	1.69	0.590013
comp	1.01	0.988384
Mean VIF	16.54	

Table 5: Results of Multicollinearity Test

Looking at the output, we see that the following variables have a VIF > 10 value: `la`, `temp_la`, `economy`, `chi`, `day_year`, and `temp_ny`. We first, drop `la`, and re-run the model. Second, we drop `ny`, and re-run the model. Third, we drop `economy` and re-run the model. Finally, we drop `chi` and we end up with variables with a VIF < 10 score.

3.4 Test for Non-linearity

Table 6 shows the `ovtest` results for Non-linearity. The p-value is > 0.05, so we do not need a higher power X in the model.

```
Ramsey RESET test using powers of the fitted values of salespercap
       Ho:  model has no omitted variables
                 F(3, 987) =      2.09
                  Prob > F =      0.1001
```

Table 6: Non-linearity test results

3.5 Test for Heteroskedasticity

Table 7 shows the test results for Heteroskedasticity with `hettest` command

```
Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
         Ho: Constant variance
         Variables: fitted values of salespercap

         chi2(1)      =     5.50
         Prob > chi2  =   0.0190
```

Table 7: Heteroskedasticity test results

The null hypothesis for the test is that our residuals have constant variance (i.e.it is homoskedastic). The p-value is < 0.05, we reject the null hypothesis and conclude the residuals are heteroskedastic.

3.6 Test for Normality & Outliers

Testing `sktest res` we find a P-value of 0.0001, so we reject the Null and conclude that the residual `res` is not normally distributed. We evaluate standard residual `sres` and we drop outliers with `sres` values smaller than -2.0 and bigger than 2.0. This results in a new `sktest res` p-value of 0.3102 hence our data is now normally distributed. Figure 5 and Figure 6 show the results of the normality tests without outliers.

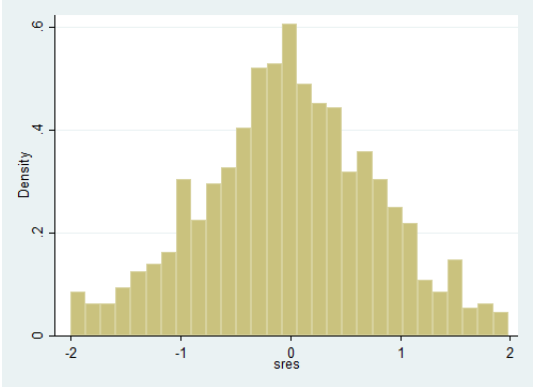

Figure 5: Density distribution of Standard Residual

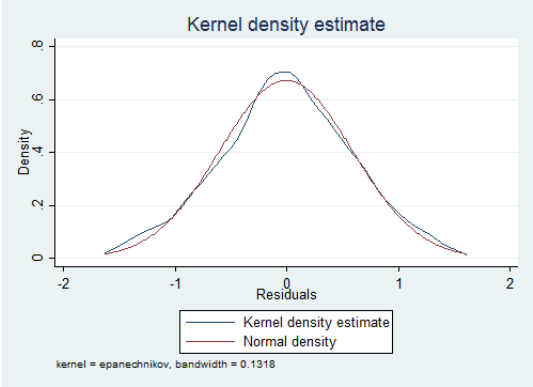

Figure 6: Normality Test on Residuals

3.7 Obtaining Residuals

We check the residuals with `rvfplot` in Figure 7 to check if there is a weird distribution of the data

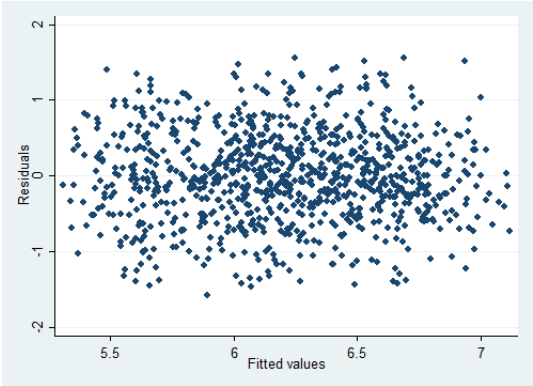

Figure 7: Residual plot distribution

The data looks randomly distributed and no weird shape.

3.8 Cook's Distance & Outliers

A Cook's Distance is calculated for each row in our dataset, we see extreme values of Cook's Distance which indicate points that are probably influential in Figure 8.

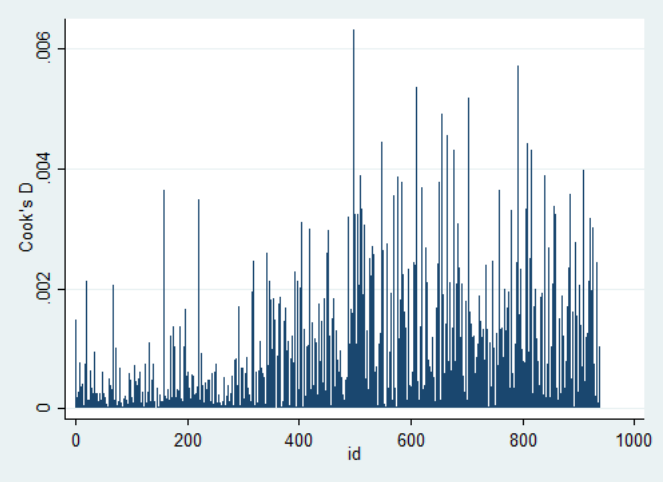

Figure 8: Result of Cook's Distance

We drop observations (outliers) with Cook's Distance $D > 0.005$

3.9 Regular Backward Regression

We implement at regular backward regression (throwing out highest p-value one at a time) with the following X variables: `price`, `comp`, `day_year`, `day_week`, `weekend`, `newchain`, `temp_ny`, `temp_chi`, `temp_la`, and `promo`. Table 8 shows the new model (Model II) with price as baseline

```
                    begin with full model
p = 0.5937 >= 0.0500  removing day_week
p = 0.4920 >= 0.0500  removing newchain
p = 0.4635 >= 0.0500  removing day_year
```

Source	SS	df	MS	Number of obs	=	932
				F(6, 925)	=	73.82
Model	152.848733	6	25.4747888	Prob > F	=	0.0000
Residual	319.221316	925	.345104125	R-squared	=	0.3238
				Adj R-squared	=	0.3194
Total	472.070049	931	.507056981	Root MSE	=	.58746

salespercap	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
comp	.6847731	.1347475	5.08	0.000	.420327	.9492193
temp_chi	-.0065038	.0016045	-4.05	0.000	-.0096526	-.003355
promo	.3933279	.0435895	9.02	0.000	.307782	.4788737
weekend	.2949089	.0427796	6.89	0.000	.2109526	.3788653
temp_la	.0048328	.0011364	4.25	0.000	.0026025	.0070631
temp_ny	.0056636	.0014659	3.86	0.000	.0027867	.0085404
_cons	4.95907	.1811533	27.37	0.000	4.603551	5.31459

Table 8: Model II - Regular Backward Regression Model

From this result, we don't need the variables `day_week`, `newchain`, and `day_year` in the model.

3.10 Backward Regression using Adjusted R-Square

Our next model implements `backward r2adj` and optimizes for Adjusted R-Square. Table 9 shows the results of our next model (Model III).

Source	SS	df	MS	Number of obs	=	932
				F(6, 925)	=	73.82
Model	152.848733	6	25.4747888	Prob > F	=	0.0000
Residual	319.221316	925	.345104125	R-squared	=	0.3238
				Adj R-squared	=	0.3194
Total	472.070049	931	.507056981	Root MSE	=	.58746

salespercap	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
comp	.6847731	.1347475	5.08	0.000	.420327	.9492193
weekend	.2949089	.0427796	6.89	0.000	.2109526	.3788653
temp_ny	.0056636	.0014659	3.86	0.000	.0027867	.0085404
temp_chi	-.0065038	.0016045	-4.05	0.000	-.0096526	-.003355
temp_la	.0048328	.0011364	4.25	0.000	.0026025	.0070631
promo	.3933279	.0435895	9.02	0.000	.307782	.4788737
_cons	4.95907	.1811533	27.37	0.000	4.603551	5.31459

Table 9: Model III - Backward using Adjusted R-Square Model

3.11 Backward Regression using AIC

Our final model implements `backward aic` and optimizes for Adjusted R-Square. Table 10 shows the results of our final model (Model IV).

Source	SS	df	MS	Number of obs	=	932
				F(6, 925)	=	73.82
Model	152.848733	6	25.4747888	Prob > F	=	0.0000
Residual	319.221316	925	.345104125	R-squared	=	0.3238
				Adj R-squared	=	0.3194
Total	472.070049	931	.507056981	Root MSE	=	.58746

salespercap	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
comp	.6847731	.1347475	5.08	0.000	.420327	.9492193
weekend	.2949089	.0427796	6.89	0.000	.2109526	.3788653
temp_ny	.0056636	.0014659	3.86	0.000	.0027867	.0085404
temp_chi	-.0065038	.0016045	-4.05	0.000	-.0096526	-.003355
temp_la	.0048328	.0011364	4.25	0.000	.0026025	.0070631
promo	.3933279	.0435895	9.02	0.000	.307782	.4788737
_cons	4.95907	.1811533	27.37	0.000	4.603551	5.31459

Table 10: Model IV - Backward Regression using AIC Model

4 Conclusion & Discussion

We determine, using different diagnostic tools, what effects are contained in the dataset help to explain CrimsonCream Inc ice cream sales. We run 4 different regression models and the results are show in Table 11

Model	Regression Type	R^2	Adjusted R^2	S_e
Model I	Naïve	0.2598	0.2501	0.7414
Model II	Normal Backward	0.3238	0.3194	0.5874
Model III	Backward Adjusted R^2	0.3238	0.3194	0.5874
Model IV	Backward AIC	0.3238	0.3194	0.48746

Table 11: Summary of Regression Models

Model IV has the lowest S_e value, hence we present the equation that adequately describes the sales per capita with price as baseline:

```
salespercap = 0.684*comp + 0.393*promo + 0.295*weekend - 0.006*temp_chi +
                0.005*temp_la + 0.005*temp_ny + 4.95
```

We observe that the Adjusted R-Square value of Model IV higher and the value of S_e is lower compared to Model I which indicates this is a better model than our naïve regression model. Looking at the S_e, if we use this model to predict the core price of ice cream sales, we would be at +/- 0.96 units with 95% confidence.

This formula takes on two forms for `promo = 0,1`.

- For `promo = 0`

```
y = 0.684*comp + 0.295*weekend - 0.006*temp_chi + 0.005*temp_la +
                0.005*temp_ny + 4.95
```

- For `promo = 1`

```
y = 0.684*comp + 0.295*weekend - 0.006*temp_chi + 0.005*temp_la +
                0.005*temp_ny + 5.34
```

For a given day (`weekend` = 0,1), the sales on promo (`promo` = 1), have on average sales per capita 0.393 points higher than those not running on promo (`promo` = 0). This shows that the promotional Marketing campaign worked. There was more sales when the campaign was running.

References

[1] StataCorp. 2015. Stata Statistical Software: Release 14. College Station, TX: StataCorp LP.

[2] STAT 104 Quantitative Methods for Economists. 2016. Dataset taken from http://people.fas.harvard.edu/∼mparzen/stat104/icecream2016V1